“Testing” for Advanced Networking Technologies

Craig Hunt
Senior Technical Advisor
craig.hunt@nist.gov
1-301-975-3827

Doug Montgomery
Manager Internetworking Technologies
dougm@nist.gov
1-301-975-3630

Advanced Network Technologies Division
Information Technology Laboratory

NIST

History of NIST’s Efforts in Networking Technology Testing

- **Traditional Focus of Network Testing**
  - **Goal:**
    - Validation of commercially available end products
  - **Customer:**
    - Testing to support end consumers / procurement
  - **Activities:**
    - Conformance testing, standardized test suites, certification of test centers, formal specification
- **NIST’s Rich History of Contributions**
  - ATM Forum Conformance Testing (ABR TM, PNNI)
  - MPEG Conformance Testing
  - ISDN Testing
  - OSI Testing & GOSIP Test Program
  - Formal Methods, Conformance Test System Research

Shifting the Focus of Test and Measurement

- **Evolving Focus of Network Testing**
  - **Goal:**
    - ⁎ Expedite the research, development, standardization, and commercialization of high risk technologies
    - ⁎ Support the <u>early development</u> of new technology with broad, industry building potential
    - ⁎ Test and measure early enough to <u>impact the design</u> of new technology
  - **Customer:**
    - ⁎ Testing to support industry technology developers and researchers.
  - **Activities:**
    - ⁎ Rapid prototyping, interoperability testing, simulation and modeling, performance testing, light weight test tools, experimentation tools

The Push and Pull of Testing

- **Pushing Testing Technology**
  - Traditional validation / conformance efforts pushed by user groups
  - Required by procurement processes
  - Vendors view testing as a necessary / required evil
  - Testing may take on a life of its own, becoming a hindrance to technology development / adoption
- **Creating Pull for Testing Technology**
  - Testing as a voluntary service made available to industry
  - Testing must provide a tangible value to developers
  - Testing technology and tests must be easily usable and available
  - Tests and tools must be available very early in product life cycle
    - “*Good enough*” testing early in the design and development cycle
  - “Testing” often involves rapid prototyping and analysis of designs

Creating a Pull for Testing

- **Choosing targets**
  - Deciding where and how to apply testing technology
  - Surfing the technology waves ….
    - Industry must be committed to solving problems / building products
      - NIST doesn’t create the waves
    - NIST’s work must be early enough to impact design and expedite the commercial development of first products
      - NIST’s works with industry to expedite the development of the wave and influence its shape / direction
    - As design / standards mature and first products are released
      - The wave has broke and it is too late to do “testing” on the leading edge
      - Typical rides on technology waves are 6 to 36 months.

Creating Pull for Testing

- **Creating Testing Technology that People Want to Use**
  - Usability and availability of testing technology is key to its success
    - Test tools must make minimal requirements on the system under test
      - don’t require / expect test harnesses and interfaces
    - Test tools must be usable by product research and development staff
      - a more comprehensive test tool that is too complex won’t be used
    - Test tools must be portable or remotely accessible
      - exploit ubiquitous WWW technology (remote access, applets)
      - exploit publicly / commonly available hardware/software platforms

Creating Pull for Testing

- **Good Enough Testing**
  - Goal is to expedite design and first development
    - Not to certify final product
  - Lightweight / usable systems that address 80% of testing requirements
    - Don’t focus on negative testing
  - Assume test system users are very knowledgeable about the technology
    - Test systems & languages can produce diagnostics rather than verdicts

Types of “Testing” Activities

- **Performance Testing**
  - Tools and techniques to assist in the measurement and characterization of networking technologies, middleware, distributed systems, and hardware components
  - Focus is on the development of the tools and techniques
  - Characterization of general technology not benchmarking specific products
- **Rapid Prototyping / Reference Implementations**
  - Feasibility prototypes of emerging technologies / standards
  - Must be done very early in design / standardization cycle
  - NIST prototypes / reference implementations serve several purposes:
    - testing feasibility of designs
    - improving the quality of specifications / standards
    - providing publicly available basis for expediting commercial development
    - establishing a reference / basis for testing other implementations

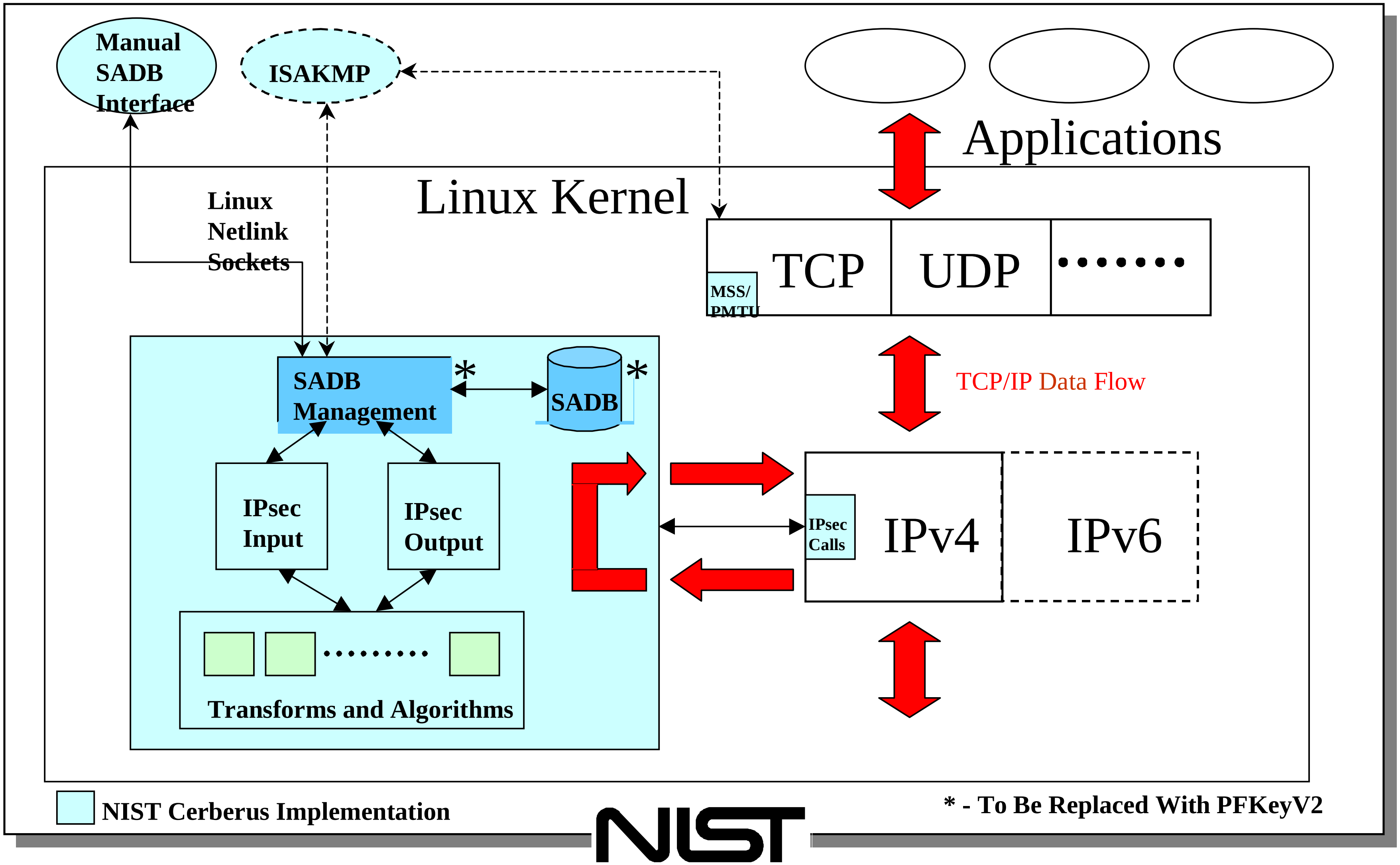
More Types of “Testing”

- **Interoperability Testing**
  - Methodologies and tools to facilitate interoperability testing & pilots
  - Interoperability testing focused on early stages of product R&D
  - NIST’s roles in industry interoperability testing:
    - providing reference implementations and testing tools
    - facilitating multi-vendor interoperability testing events / testbeds
- **Test & Measurement Research**
  - Advanced methodologies to improved the capabilities and quality of testing and measurement in areas such as:
    - Evaluating collaboration environments
    - Testing distributed, topologically sensitive, multi-party protocols
    - Fine grained instrumentation of protocol implementations
- **Some Examples of our work ……**

Rapid Prototyping: NIST Cerberus

- ***Leading edge prototype of IETF IPSec technology***
  - Reference prototype implemented as Linux kernel modules
  - Platform for further NIST research in advanced IPSec issues
  - Public basis for interoperability testing, pilot deployment & research
    - 4 software releases since Q4 1997
    - Used by ~150 organizations in industry, government and academia
- **Cerberus:**
  - Full implementation of IETF IPv4 AH and ESP security protocols
    - Supports both Host and Router IPSec functions
    - Supports large library of security transforms / crypto algorithms
    - Integrated with *NIST Pluto++* Internet Key Exchange (IKE) prototype
    - Core component of IPSec-WIT on-line test system
  - Future research and development
    - policy management, IPv6, integration with IKE / PKIX, mobility, multicast

Cerberus Software Architecture



NIST Cerberus Implementation

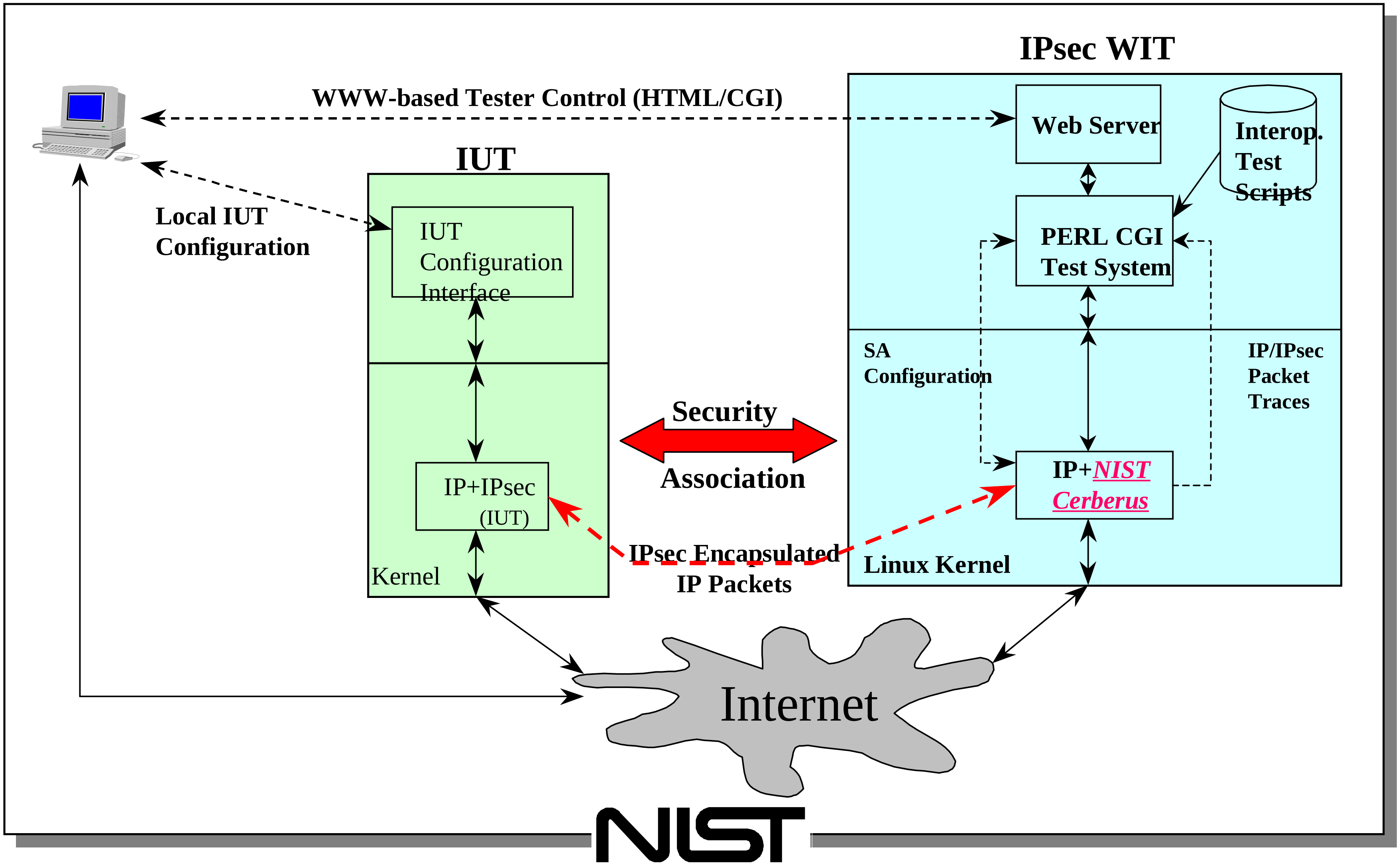
* - To Be Replaced With PFKeyV2

Interoperability Testing: IPSec-WIT

- ***IPSec / IKE / PKIX Interoperability testing anytime, anywhere***
  - Asynchronous interoperability testing … test at your own rate
  - Test without relocating system under test, or test system
- **IPSec WWW-based Interoperability Tester**
  - Practical testing tool - semi-automated, WWW driven tester
  - Built around NIST Cerberus IPSec and Pluto++ IKE implementations
  - WWW forms to navigate interoperability test scripts (~400)
  - WWW / CGI driven configuration of Cerberus / Pluto++ prototype
  - WWW-based / email examination of test results
- **Future IPSec-WIT and WWW-based Interoperability Testing**
  - Expand current system to PKIX, IPv6 - IPSec
  - Tools for automated test suite generation, extensible test languages
  - Port IPsec-WIT test engine to other IPsec implementations
  - Explore use of other WWW technology in on-line test systems

NIST

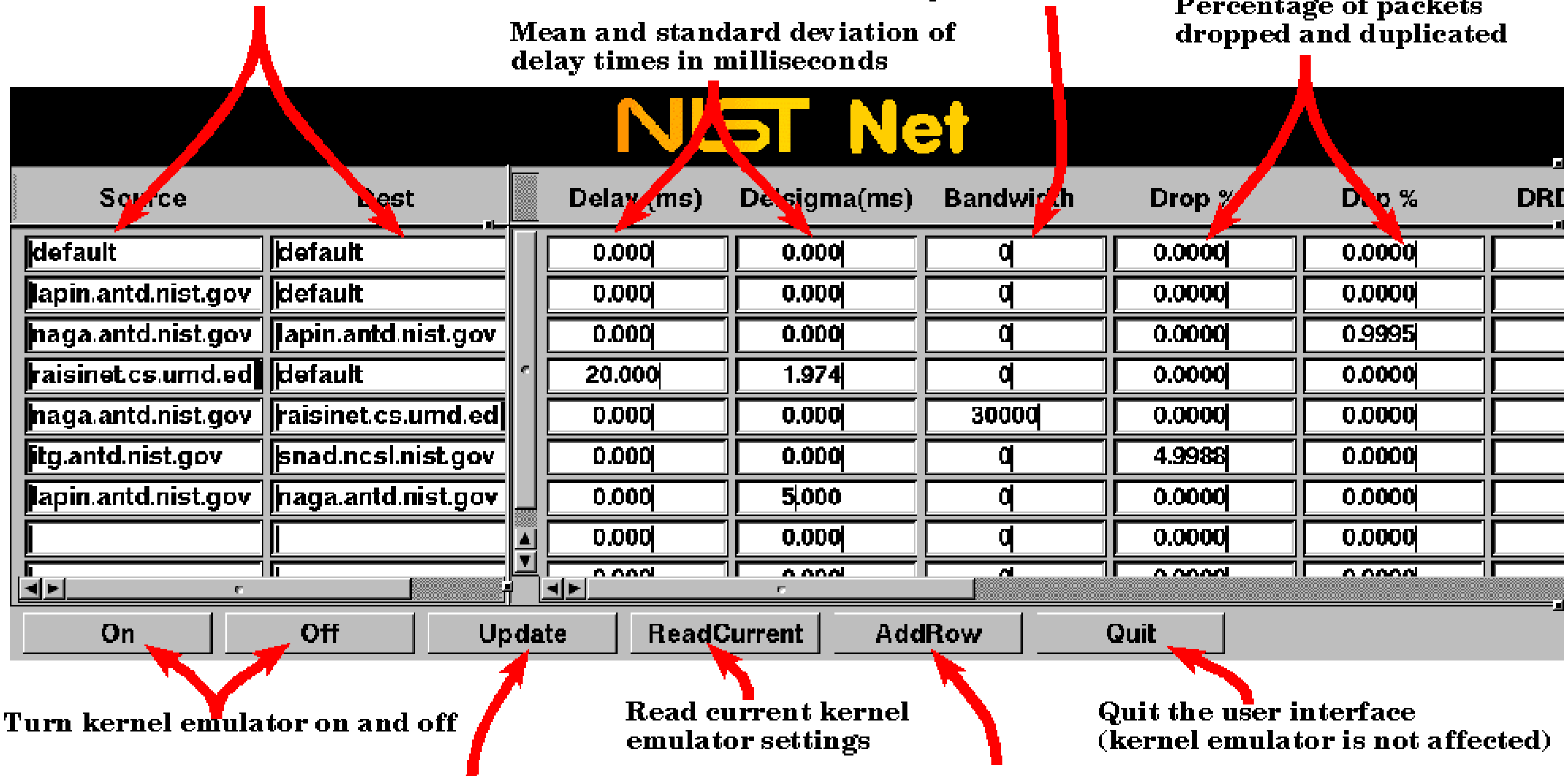
IPsec WIT System Architecture



Performance Measurement: NIST Net

- **NIST Network Emulation Tool**
  - General purpose “Internet cloud” performance emulator
  - Platform - Linux kernel loadable modules on PC hardware
  - Enables controlled, laboratory emulation of IP performance dynamics
    - Bandwidth limitations (including asymmetric bandwidth scenarios)
    - Delay distributions (jitter)
    - Packet loss, corruption, reordering
    - Router congestion avoidance schemes (e.g. DRD)
  - Supports interactive interface or parameterization from trace files
- **Applications**
  - QoS sensitivity testing, ICV Evaluations, VOIP testing
- **Future Directions**
  - Use of emulation in large scale hybrid simulations & test environments.

NIST Net



Packet source and destination addresses
(default matches all otherwise unmatched)
Either names or IP addresses may be used.

Mean and standard deviation of
delay times in milliseconds

Maximum allowed bandwidth
in bytes/second

Percentage of packets
dropped and duplicated

Source	Dest	Delay(ms)	Delsigma(ms)	Bandwidth	Drop %	Dup %	DRD
default	default	0.000	0.000	0	0.0000	0.0000	
lapin.antd.nist.gov	default	0.000	0.000	0	0.0000	0.0000	
naga.antd.nist.gov	lapin.antd.nist.gov	0.000	0.000	0	0.0000	0.9995	
raisinet.cs.umd.ed	default	20.000	1.974	0	0.0000	0.0000	
naga.antd.nist.gov	raisinet.cs.umd.ed	0.000	0.000	30000	0.0000	0.0000	
itg.antd.nist.gov	snad.ncsl.nist.gov	0.000	0.000	0	4.9988	0.0000	
lapin.antd.nist.gov	naga.antd.nist.gov	0.000	5.000	0	0.0000	0.0000	
		0.000	0.000	0	0.0000	0.0000	
		0.000	0.000	0	0.0000	0.0000	

Turn kernel emulator on and off

Load changed settings
into kernel emulator

Read current kernel
emulator settings

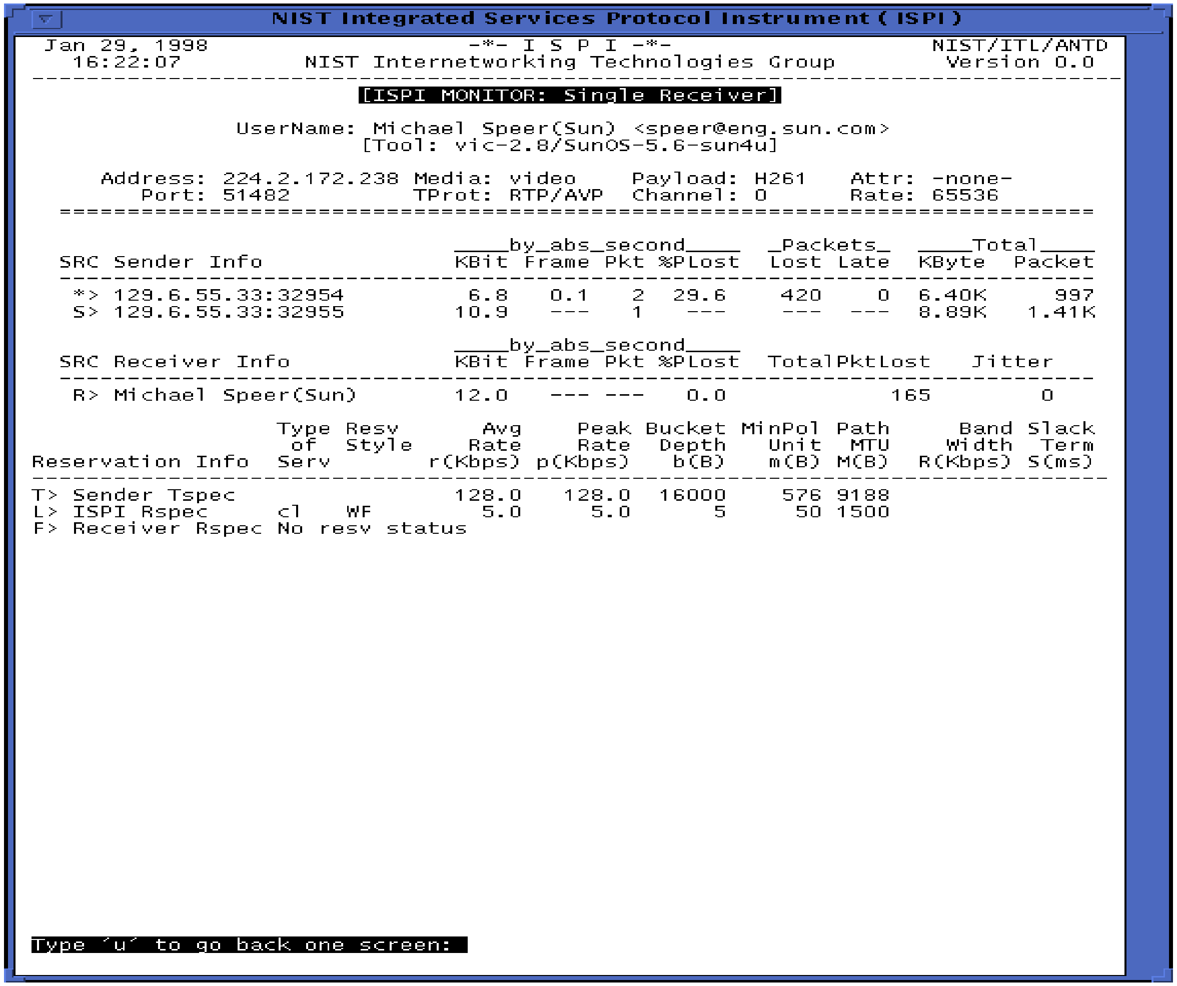
Add another row to
the user interface

Quit the user interface
(kernel emulator is not affected)

Performance / Experimentation: ISPI

- **NIST ISPI - Integrated Services Protocol Instrument**
  - Measurement / experimentation tool for IP QoS protocols
  - Support of protocol engineering and application experimentation
  - ISPI is “smart” about: RTP/RTCP, RSVP, IP Multicast, SDR protocols
- **ISPI Capabilities**
  - Monitors multicast (SDR) sessions
  - Computes performance of real time (RTP) streams
    - Measures local RTP performance
    - Reports remote RTCP performance
  - Enables proxy resource reservation (RSVP) experiments
  - Provides diagnostic traces of reservation state in network
- **Future Directions**
  - ISPI enhancements - RSVP Diagnostics, RTSP, Multicast Routing,
  - Advanced Testing technology for RSVP integrated services

NIST ISPI - RTP/RSVP Performance



```
 Jan 29, 1998                   -*- I S P I -*-                   NIST/ITL/ANTD
   16:22:07         NIST Internetworking Technologies Group        Version 0.0
--------------------------------------------------------------------------------
                        [ISPI MONITOR: Single Receiver]

               UserName: Michael Speer(Sun) <speer@eng.sun.com>
                        [Tool: vic-2.8/SunOS-5.6-sun4u]

     Address: 224.2.172.238 Media: video    Payload: H261   Attr: -none-
        Port: 51482         TProt: RTP/AVP  Channel: 0      Rate: 65536
  ============================================================================

                               ____by_abs_second____  _Packets_  ____Total____
  SRC Sender Info              KBit Frame Pkt %PLost  Lost Late  KByte  Packet
  ----------------------------------------------------------------------------
   *> 129.6.55.33:32954         6.8   0.1   2  29.6    420    0  6.40K     997
   S> 129.6.55.33:32955        10.9   ---   1   ---    ---  ---  8.89K   1.41K

                               ____by_abs_second____
  SRC Receiver Info            KBit Frame Pkt %PLost  TotalPktLost   Jitter
  ----------------------------------------------------------------------------
   R> Michael Speer(Sun)       12.0   --- ---   0.0            165        0

                  Type Resv      Avg    Peak Bucket MinPol Path     Band Slack
                   of  Style     Rate   Rate  Depth   Unit  MTU    Width  Term
Reservation Info  Serv       r(Kbps) p(Kbps)   b(B)   m(B) M(B)  R(Kbps) S(ms)
-------------------------------------------------------------------------------
T> Sender Tspec                128.0   128.0  16000    576 9188
L> ISPI Rspec     cl   WF        5.0     5.0      5     50 1500
F> Receiver Rspec No resv status











Type ´u´ to go back one screen:
```

Rapid Prototyping:

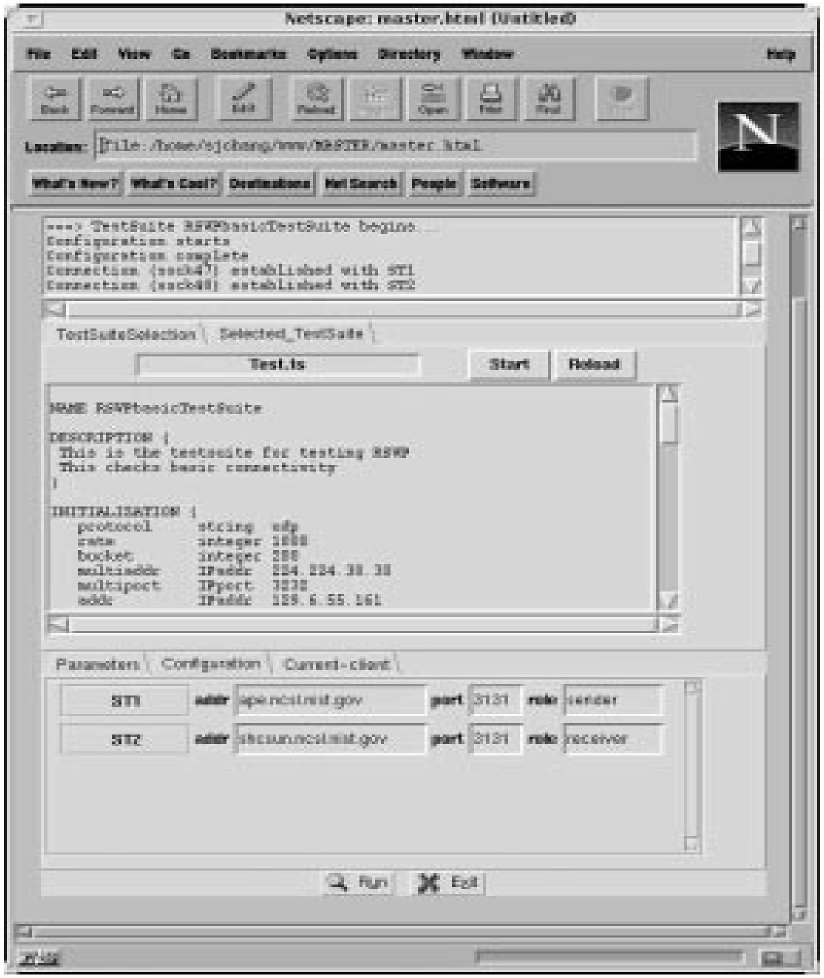
NIST Switch

- **IP Quality of Service:** ***There is no silver bullet ….***
  - Fundamental goal of NGI research program
  - Many new technologies / approaches to IP QoS
    - IP / ATM, RSVP / IntServ, DiffServ, MPLS
  - MPLS provides mechanisms to leverage many QoS issues
- **NIST Switch - IP QoS Research Platform**
  - Public domain, modular platform for MPLS research & development
  - Supports: label switching, LDP, RSVP, DiffServ, ATM
  - Current Focus:
    - Integrated MPLS-based forwarding
    - Use of RSVP to support MPLS, explicit QoS Routing
  - Future Work:
    - New QoS and load-balancing routing algorithms
    - Heterogeneous IP QoS end-to-end environments

Testing Methodology Research: DIPPER

- **Distributed Internet Protocol and PERformance Test System**
  - Testing and experimentation with IP QoS hindered by lack of tools
  - Need to test topologically sensitive signaling protocols
  - Test system portability to support field testing
- **DIPPER:**
  - Distributed test tool for protocol functionality / performance testing
    - Initial focus: RSVP / IntServ and NIST Switch testing
  - Multi-party distributed system built on top of UCB MASH Toolkit
  - Object oriented (O-Tcl and C++) test system design
  - Test system components and test scripts run in WWW browser
  - Master Tester - single point of control of distributed testers
  - O-TCL based distributed programming protocol test language
- **Future directions**
  - Use of mobile agents, application of new WWW technology to testing

DIPPER


Netscape: master.html (Untitled)
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Edit Reload Images Open Print Find Stop
What's New? What's Cool? Destinations Net Search People Software
TestSuiteSelection Selected_TestSuite
Start Reload
Parameters Configuration Current-client
Run Exit

Location: file:/home/sjchang/www/MASTER/master.html

```
===> TestSuite RSVPbasicTestSuite begins ...
Configuration starts
Configuration complete
Connection {sock47} established with ST1
Connection {sock48} established with ST2
```

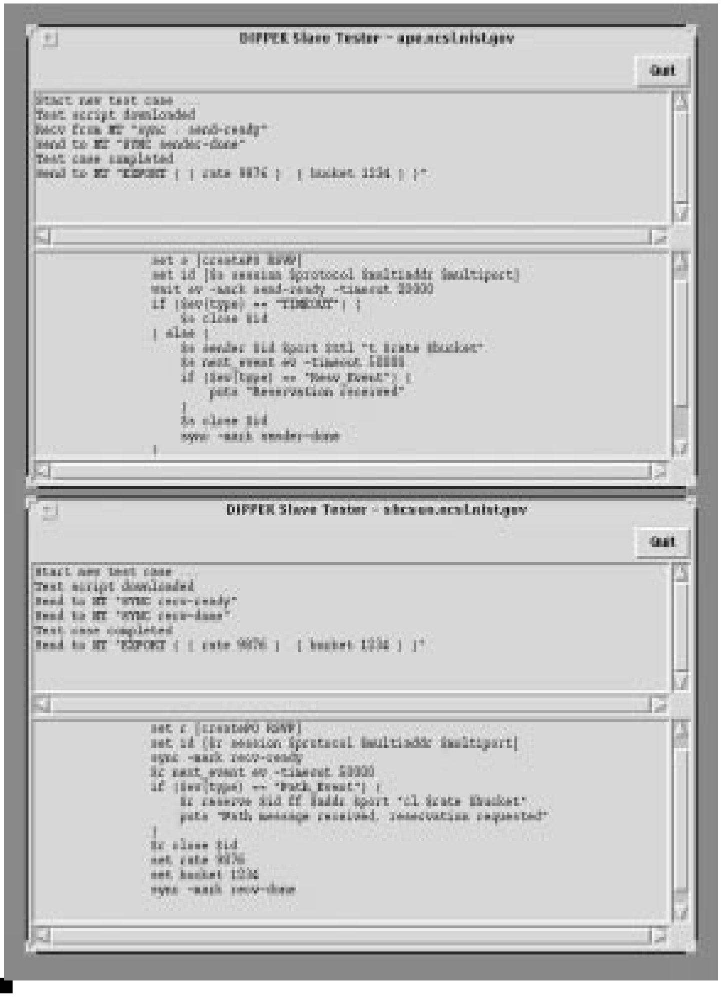
Test.ts

```
NAME RSVPbasicTestSuite

DESCRIPTION {
 This is the testsuite for testing RSVP
 This checks basic connectivity
}

INITIALISATION {
   protocol      string  udp
   rate          integer 1000
   bucket        integer 200
   multiaddr     IPaddr  224.224.30.30
   multiport     IPport  3230
   addr          IPaddr  129.6.55.161
```

ST1	addr	ape.ncsl.nist.gov	port	3131	role	sender
ST2	addr	shcsun.ncsl.nist.gov	port	3131	role	receiver


DIPPER Slave Tester - ape.ncsl.nist.gov
Quit
DIPPER Slave Tester - shcsun.ncsl.nist.gov
Quit

```
Start new test case ...
Test script downloaded
Recv from MT "sync : send-ready"
Send to MT "SYNC sender-done"
Test case completed
Send to MT "EXPORT { { rate 9876 }  { bucket 1234 } }"
```

```
set s [createPS RSVP]
set id [$s session $protocol $multiaddr $multiport]
wait ev -mark send-ready -timeout 20000
if {$ev(type) == "TIMEOUT"} {
    $s close $id
} else {
    $s sender $id $port $ttl "t $rate $bucket"
    $s next_event ev -timeout 10000
    if {$ev(type) == "Resv Event"} {
        puts "Reservation received"
    }
    $s close $id
    sync -mark sender-done
}
```

```
Start new test case ...
Test script downloaded
Send to MT "SYNC recv-ready"
Send to MT "SYNC recv-done"
Test case completed
Send to MT "EXPORT { { rate 9876 }  { bucket 1234 } }"
```

```
set r [createPU RSVP]
set id [$r session $protocol $multiaddr $multiport]
sync -mark recv-ready
$r next_event ev -timeout 50000
if {$ev(type) == "Path Event"} {
    $r reserve $id ff $addr $port "cl $rate $bucket"
    puts "Path message received, reservation requested"
}
$r close $id
set rate 9876
set bucket 1234
sync -mark recv-done
```

For More Information ...

- NIST
  - http://www.nist.gov/
- NIST / Information Technology Laboratory (ITL)
  - http://www.itl.nist.gov/
- NIST / ITL/ Advanced Networks Technologies Division (ANTD)
  - http://www.antd.nist.gov/
- NIST / ITL / ANTD / Internetworking Technologies Group (ITG)
  - http://www.antd.nist.gov/itg/
- Next Generation Internet (NGI) Program
  - http://www.ngi.gov/
- Internet Engineering Task Force (IETF)
  - http://www.ietf.org